# По-сложни проверки

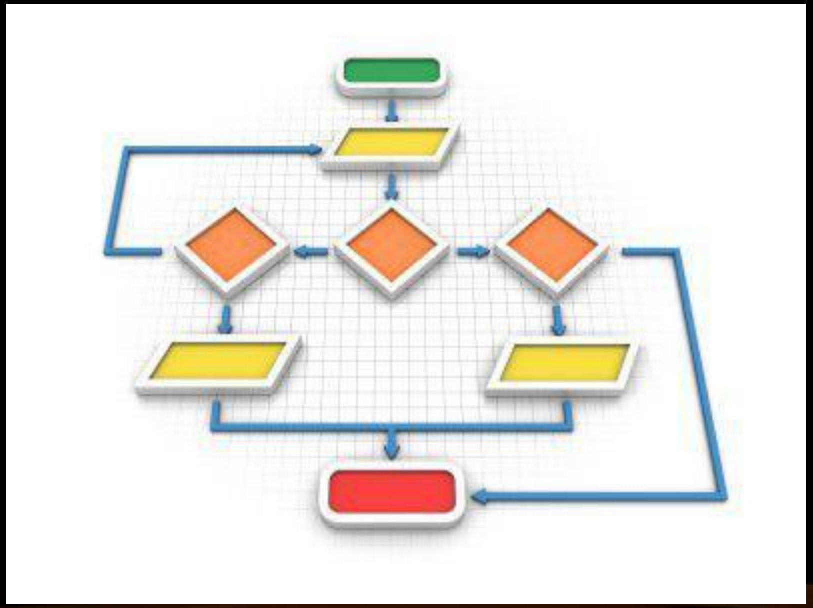
## Вложени if конструкции и по-сложни логически условия


SOFTWARE UNIVERSITY FOUNDATION










**СофтУни**

**трейнърски екип**

**Софтуерен университет**

http://softuni.bg

# Have a Question?

sli.do

#TODO

# Съдържание

1. Вложени проверки
2. По-сложни проверки
   - Логическо "**и**", "**или**", **отрицание** и **приоритет на условия**
3. Switch-case конструкция





# Вложени проверки

If-конструкции, вложени една в друга

# Вложени проверки

- Конструкциите **if-else** могат да се влагат една в друга:

```
if (condition1) {
      cout << "condition1 valid" << endl;
   if (condition2)
      cout << "condition2 valid" << endl;
   else
      cout << "condition2 not valid" << endl;
}
```

**Вложена if конструкция**

- Само при изпълнение на първото условие се преминава към вложената проверка

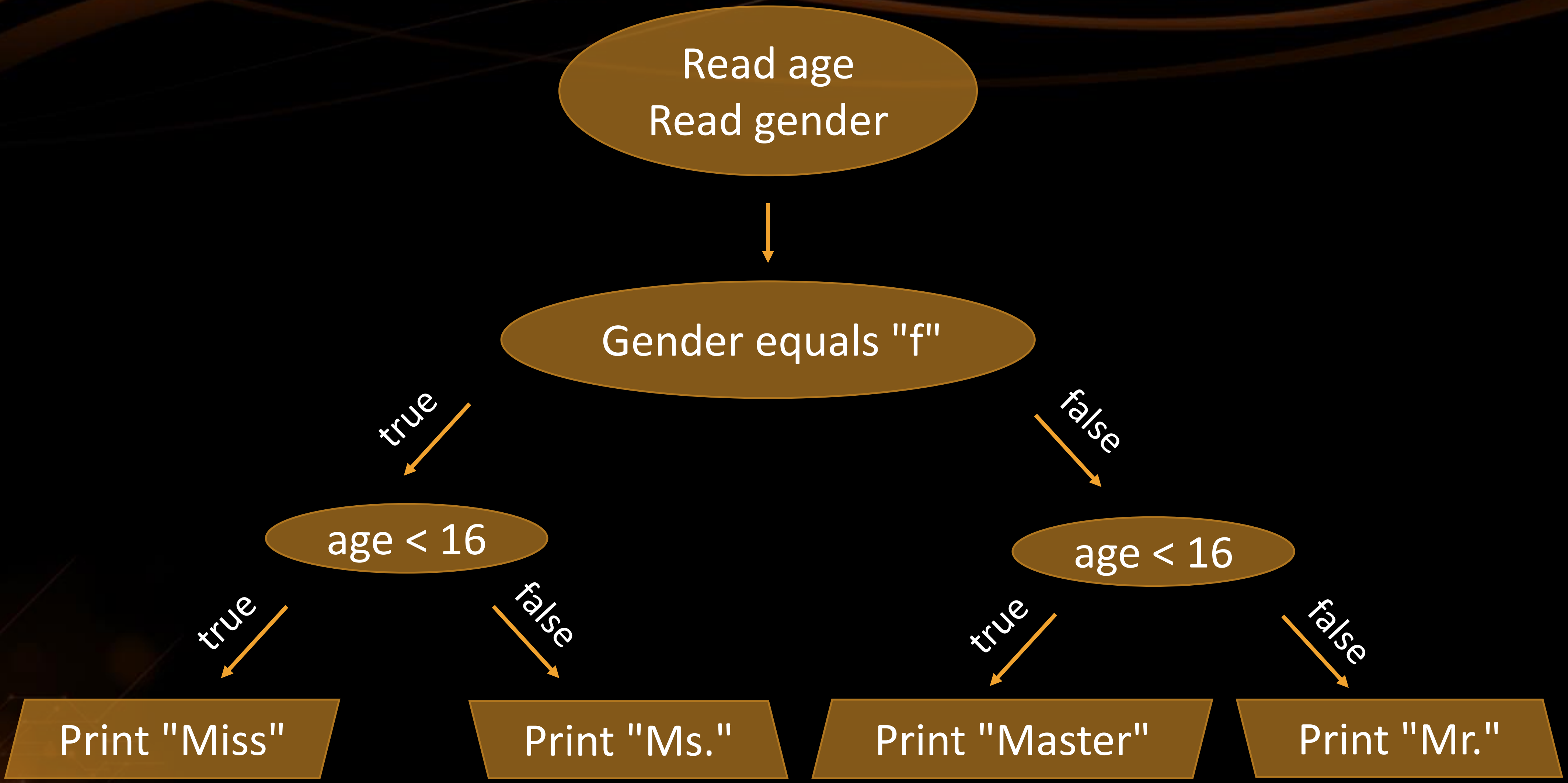
# Обръщение според възраст и пол – условие

- Напишете програма, която:
  - Чете от потребителя:
    - Възраст
    - Пол
  - Принтира обръщение според въведеното, както е показано на схемата (в следващия слайд)
- Примерен вход и изход:

```
12
f
```
→
```
Miss
```

```
16
m
```
→
```
Mr.
```

# Обръщение според възраст и пол – условие (2)



# Решение: Обръщение според възраст и пол

```
#include <iostream>
#include <string>
using namespace std;

int main() {
      double age;
      string gender;
      cin >> age >> gender;
```

Тестване на решението: [https://judge.softuni.bg/Contests/Practice/Index/531#0](https://judge.softuni.bg/Contests/Practice/Index/531#0)

```
if (gender == "m") {
   if (age < 16)
      cout << "Master";
   else if (age >= 16)
      cout << "Mr.";
 }
 else if (gender == "f")
 {
   if (age < 16)
      cout << "Miss";
   else if (age >= 16)
      cout << "Ms.";
 }
}
```

Тестване на решението: [https://judge.softuni.bg/Contests/Practice/Index/531#0](https://judge.softuni.bg/Contests/Practice/Index/531#0)

# Квартално магазинче – условие

- Напишете програма, която:
  - Чете от потребителя:
    - Име на продукт
    - Град
    - Количество
  - Пресмята цената му спрямо таблицата:

| град / продукт | coffee | water | beer | sweets | peanuts |
|---|---|---|---|---|---|
| Sofia | 0.50 | 0.80 | 1.20 | 1.45 | 1.60 |
| Plovdiv | 0.40 | 0.70 | 1.15 | 1.30 | 1.50 |
| Varna | 0.45 | 0.70 | 1.10 | 1.35 | 1.55 |

# Квартално магазинче – условие (2)

- Примерен вход и изход:

| Вход | Изход |
| --- | --- |
| coffee<br>Varna<br>2 | 0.9 |
| peanuts<br>Plovdiv<br>1 | 1.5 |
| beer<br>Sofia<br>6 | 7.2 |

# Решение: Квартално магазинче

```cpp
#include <iostream>
#include <string>
using namespace std;

int main() {
    string product, town;
    double quantity;
    cin >> product >> town >> quantity;
```

Тестване на решението: https://judge.softuni.bg/Contests/Practice/Index/531#1

# Решение: Квартално магазинче (2)

```
if (town == "Sofia"){
      if (product == "coffee")
            cout << quantity * 0.5 << endl;
      // TODO: finish the checks for all the products…
else if(town == "Varna")
     // TODO: finish the checks for all the products…
else if(town == "Plovdiv")
    // TODO: finish the checks for all the products…
```

Тестване на решението: [https://judge.softuni.bg/Contests/Practice/Index/531#1](https://judge.softuni.bg/Contests/Practice/Index/531#1)




# По-сложни проверки

Булеви оператори

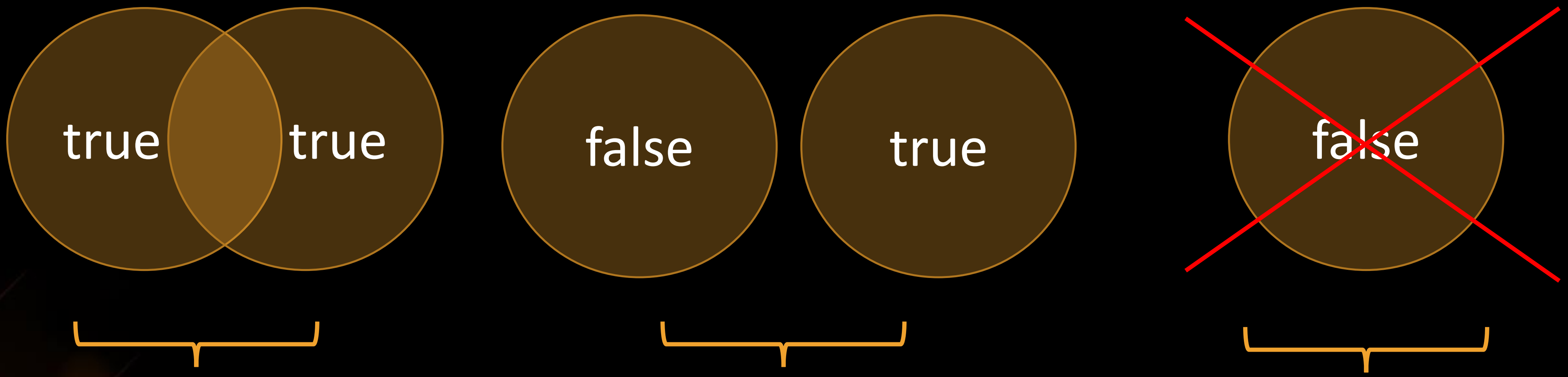
# Булеви оператори

- Оператори, които комбинират или изключват условия
- Връщат булеви резултат (true или false)

"&&" - И

"||" - ИЛИ

"!" - ОТРИЦАНИЕ




Вярност на двете условия

Вярност на едното или другото условие

Отрицание на условие

# Логическо "И"

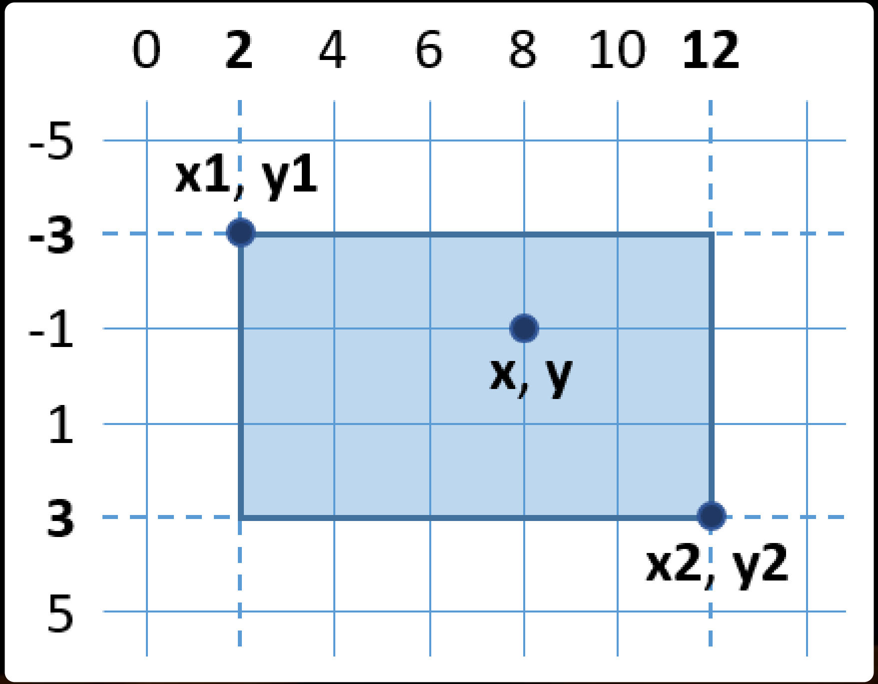
- Проверява изпълнението на няколко условия едновременно
- Пример: проверка дали число е едновременно:
  - по-голямо от 5 и по-малко от 10
  - четно

&&

```
int a;
cin >> a;
if (a > 5 && x < 10 && a % 2 == 0) …
```

# Точка в правоъгълник - условие

- Напишете програма, която:
  - Проверява дали точка е вътрешна за даден правоъгълник
- Точка е вътрешна, ако е едновременно:
  - надясно от лявата му страна
  - наляво от дясната му страна
  - надолу от горната му страна
  - нагоре от долната му страна



# Точка в правоъгълник - решение

```
double x, x1 …;
cin >> x, x1 …;
// TODO: Read the coordinates of the points …
if (x >= x1 && x <= x2 && y >= y1 && y <= y2)
    cout << "Inside" << endl;
else
    cout << "Outside" << endl;
```

Тестване на решението: https://judge.softuni.bg/Contests/Practice/Index/531#2

# Логическо "ИЛИ"

- Проверява дали е изпълнено поне едно измежду няколко условия
- Пример: проверка дали въведената дума е:
  - "Example" или "Demo"

||

#include <string>

```
string word;
cin >> word;
if (word == "Example" || word == "Demo") …
```

# Плод или зеленчук - условие

- Напишете програма, която:
  - Проверява дали въведеният вход от потребителя е плод или зеленчук измежду изброените:
    - Плодове: banana, apple, kiwi, cherry, lemon, grapes
    - Зеленчуци: tomato, cucumber, pepper, carrot
  - Извежда "vegetable","fruit" или "unknown"
- Примерен вход и изход:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| `lemon` | → | `fruit` | `carrot` | → | `vegetable` | `java` | → | `unknown` |

# Плод или зеленчук - решение

```
string food;
cin >> food;

if (food == "banana" ||
    food == "apple" ||
    food == "kiwi" ||
    food == "cherry" ||
    food == "lemon" ||
    food == "grapes")
{
  cout << "fruit" << endl;
}
```

Тестване на решението: https://judge.softuni.bg/Contests/Practice/Index/531#3

# Плод или зеленчук - решение

SOFTWARE UNIVERSITY FOUNDATION

```
else if (food == "tomato" ||
         food == "cucumber" ||
         food == "pepper" ||
         food == "carrot")
{
  cout << "vegetable" << endl;
}
else {
  cout << "unknown" << endl;
}
```

Тестване на решението: [https://judge.softuni.bg/Contests/Practice/Index/531#3](https://judge.softuni.bg/Contests/Practice/Index/531#3)

# Приоритет на условия

SOFTWARE UNIVERSITY FOUNDATION

- Чрез скоби ( ) можем да приоритизираме условия
- Пример:
  - Проверка дали число е в диапазона [100… 200] или е равно на 0

```
int a = 5;
cout << a << endl;
if ((a >= 100 && a <= 200) || a == 0) {
    cout << "In range" << endl;
}
```

# Логическо отрицание

- Проверява дали **не е изпълнено** дадено условие

!

- Пример:
  - Проверка дали число е по-голямо от 10 и е четно:

```
bool is_valid = (number > 10) && (number % 2 == 0);
if (!is_valid)
  cout << "Invalid" << endl;
```

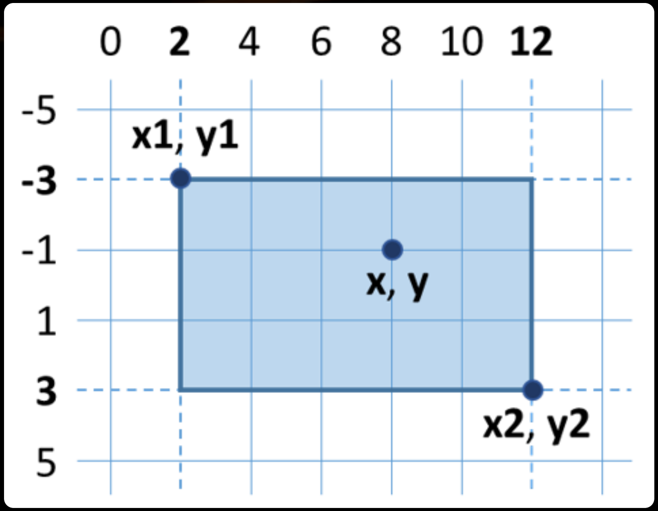



# По-сложни проверки

## Работа на живо в клас (лаб)

# Точка върху страна на правоъгълник - условие

- Напишете програма, която:
  - Чете 6 десетични числа **x1**, **y1**, **x2**, **y2**, **x** и **y**
  - Извежда дали точката е:
    - върху страна от правоъгълника ("Border")
    - в или извън правоъгълника ("Inside/Outside")




- Примерен вход и изход:

```
2
-3
12
3
8
-1
```

→

```
Inside /
Outside
```

```
2
-3
12
3
12
-1
```

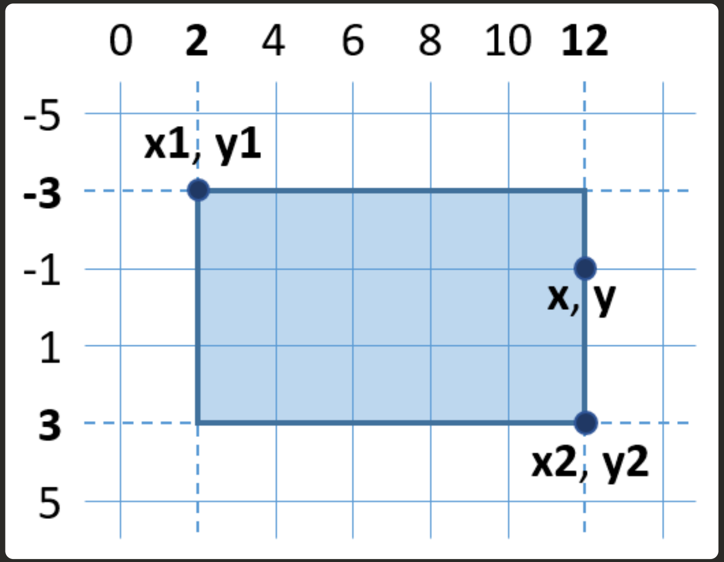
→

```
Border
```

# Точка върху страна на правоъгълник - решение

- Точка лежи **върху някоя от страните** на правоъгълник, ако:
  - **x** съвпада с **x1** или **x2** и същевременно **y** е между **y1** и **y2** или
  - **y** съвпада с **y1** или **y2** и същевременно **x** е между **x1** и **x2**

```
if (((x == x1 || x == x2) &&
     (y >= y1) && (y <= y2)) ||
    ((y == y1 || y == y2) &&
     (x >= x1) && (x <= x2))) {
   cout << "Border" << endl;
}
else {
   cout << "Inside / Outside" << endl;
}
```



# Опростяване на логически условия

- Предходното условие може да се опрости ето така:

```
bool on_left_side = (x == x1) && (y >= y1) && (y <= y2);
bool on_right_side = (x == x2) && (y >= y1) && (y <= y2);
bool on_up_side = (y == y1) && (x >= x1) && (x <= x2);
bool on_down_side = (y == y2) && (x >= x1) && (x <= x2);

if (on_left_side || on_right_side || on_up_side ||
on_down_side) {
    cout << "Border" << endl;
}
else {
   cout << "Inside / Outside" << endl; }
```

Тестване на решението: https://judge.softuni.bg/Contests/Practice/Index/531#5

# Магазин за плодове – условие

- Напишете програма, която:
  - Чете потребителски вход:
    - Продукт
    - Ден
    - Количество
  - Извежда сумата, която трябва да се заплати според деня и продукта

# Магазин за плодове – условие (2)

- В работни дни продава на следните цени:

| плод | banana | apple | orange | grapefruit | kiwi | pineapple | grapes |
|---|---|---|---|---|---|---|---|
| цена | 2.50 | 1.20 | 0.85 | 1.45 | 2.70 | 5.50 | 3.85 |

- В почивни дни цените са по-високи:

| плод | banana | apple | orange | grapefruit | kiwi | pineapple | grapes |
|---|---|---|---|---|---|---|---|
| цена | 2.70 | 1.25 | 0.90 | 1.60 | 3.00 | 5.60 | 4.20 |

- Примерен вход и изход:

```
apple
Tuesday
2
```

→

```
2.40
```

```
orange
Sunday
3
```

→

```
2.70
```

# Магазин за плодове - решение

```
if (day == "Saturday" || day == "Sunday") {
   if (fruit == "banana")
       cout << fixed << setprecision(2)
            << quantity * 2.70 << endl;
   else if (fruit == "apple")
       cout << fixed << setprecision(2)
            << quantity * 1.25 << endl;

   // TODO: check the other fruits …
}
```

Тестване на решението: [https://judge.softuni.bg/Contests/Practice/Index/531#6](https://judge.softuni.bg/Contests/Practice/Index/531#6)

# Магазин за плодове - решение (2)

```
else if (day == "Monday" ||
            day == "Tuesday" ||
            day == "Wednesday" ||
            day == "Thursday" ||
            day == "Friday")
{
   if (fruit == "banana")
      cout << fixed << setprecision(2)
            << quantity * 2.50 << endl;

   // TODO: check the other fruits …
}
```

Тестване на решението: https://judge.softuni.bg/Contests/Practice/Index/531#6

# Търговски комисионни - условие

- Напишете програма, която:
  - Чете от потребителя:
    - Град
    - Обем на продажби (десетично число)
  - Изчислява комисионната, която дадена фирма дава на търговците според града и обема на продажбите
  - Извежда стойността на комисионната, закръглена до 2 цифри след десетичната запетая

# Търговски комисионни – условие (2)

| Град / цена | 0 ≤ s ≤ 500 | 500 < s ≤ 1 000 | 1 000 < s ≤ 10 000 | s > 10 000 |
|---|---|---|---|---|
| Sofia | 5% | 7% | 8% | 12% |
| Varna | 4.5% | 7.5% | 10% | 13% |
| Plovdiv | 5.5% | 8% | 12% | 14.5% |

- Примерен вход и изход:

```
Plovdiv
499.99
```

→

```
27.50
```

# Търговски комисионни - решение

SOFTWARE UNIVERSITY FOUNDATION

```
string city;
double sales;
cin >> city >> sales;
double comission = 0;

if (city != "Sofia" &&
    city != "Varna" &&
    city != "Plovdiv" ||
    sales < 0)
{
    cout << "error" << endl;
}
```

#include <iomanip>

Тестване на решението: https://judge.softuni.bg/Contests/Practice/Index/531#7

# Търговски комисионни - решение

SOFTWARE UNIVERSITY FOUNDATION

```
else {
    if (city == "Sofia") {
        if (sales >= 0 && sales <= 500)
            comission = 0.05;
        else if (sales > 500 && sales <= 1000)
            comission = 0.07;
        // TODO: check the other price ranges …
}
```

Тестване на решението: https://judge.softuni.bg/Contests/Practice/Index/531#7

# Търговски комисионни - решение

```
    else if (city == "Varna")
        // TODO: check the price ranges …
    else if (city == "Plovdiv")
        // TODO: check the price ranges …
        cout << fixed << setprecision(2)
             << comission * sales << endl;
```

Тестване на решението: https://judge.softuni.bg/Contests/Practice/Index/531#7

# Вид животно - условие

- Напишете програма, която:
  - Чете от потребителя дума (животно)
    - Възможен вход: **"dog", "crocodile", "tortoise", "snake"**:
  - Извежда вида на животно според името му
    - Бозайник – "mammal"
    - Влечуго – "reptile"
    - Други – "unknown"
- Примерен вход и изход:

| Вход | Изход |
|---|---|
| dog | mammal |
| car | unknown |

# Вид животно - решение

```
string animal;
cin >> animal;

if (animal == "dog")
    cout << "mammal" << endl;
else if (animal == "snake" ||
         animal == "crocodile" ||
         animal == "tortoise")
    cout << "reptile" << endl;
else
    cout << "unknown" << endl;
```

Тестване на решението: https://judge.softuni.bg/Contests/Practice/Index/531#9




# Условна конструкция Switch-case

По-доброто If-Else-If-Else?

# Условна конструкция Switch-case

- Работи като поредица **`if/else-if/else-if…`**

```
switch (...){
  case …:
   //code
   break;
  case …:
   //code
   break;

  default:
   //code
   break;

}
```

Условието в switch-case е стойност

Изброяване на условия (стойности) за проверката

Код, който ще се изпълни, ако няма дефинирано условие

# Множество случаи в Switch-case

- Чрез switch-case, можем да изпълняваме един и същ код за множество условия

```
switch (...) {
  case ... :
  case ... :
  case ... :
         //код
         break;
  default:
         //код
         break;
}
```

Кодът ще се изпълни за някое от трите условия в серията

# Ден от седмицата - условие

- Напишете програма, която:
  - Чете цяло число  въведено от потребителя
  - Извежда деня от седмицата с текст (на английски) според въведеното число (1…7)
  - Извежда "Error!", ако числото не е в диапазона
- Примерен вход и изход:

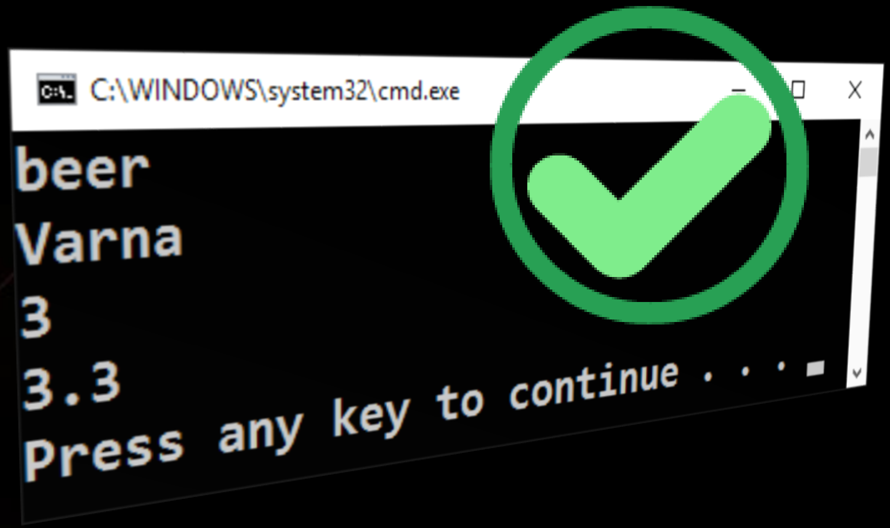
| Вход | | Изход |
|---|---|---|
| 1 | → | Monday |
| 4 | → | Thursday |

# Ден от седмицата - решение

```
int day;
cin >> day;

switch (day)
{
  case 1: cout << "Monday" << endl; break;
  case 2: cout << "Tuesday" << endl; break;
  …
  case 7: cout << "Sunday" << endl; break;
  default: cout << "Error!" << endl; break;
}
```

Тестване на решението : https://judge.softuni.bg/Contests/Practice/Index/531#8






# По-сложни проверки

Работа на живо в клас (лаб)

# Какво научихме днес?

- Вложени проверки:

```
if (condition1)
{
    if (condition2) …
    else …
}
```




- По-сложни проверки с &&, ||, ! и ()

```
if ((x == left || x == right) && y >= top && y <= bottom)
 { cout << "Point on the left or right side." << endl; }
```

# Какво научихме днес? (2)

- Switch-case конструкция

```
switch (...){
  case …:
   //code
   break;
  case …:
   //code
   break;
  default:
   //code
   break;
}
```



# По-сложни проверки



Въпроси?

https://softuni.bg/courses/programming-basics/

# Лиценз

- Настоящият курс (слайдове, примери, видео, задачи и др.) се разпространяват под свободен лиценз "Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International"

- Благодарности: настоящият материал може да съдържа части от следните източници
  - Книга "Основи на програмирането със C#" от Светлин Наков и колектив с лиценз CC-BY-SA

# Безплатни обучения в СофтУни

- Фондация "Софтуерен университет" – softuni.org
- Софтуерен университет – качествено образование, професия и работа за софтуерни инженери
  - softuni.bg
- СофтУни @ Facebook
  - facebook.com/SoftwareUniversity
- СофтУни форуми – forum.softuni.bg

